



Reputation-based content dissemination for user generated wireless podcasting

Hu, Liang; Dittmann, Lars; Le Boudec, J.-Y.

Published in:
Proceedings, WCNC

Link to article, DOI:
[10.1109/WCNC.2009.4917944](https://doi.org/10.1109/WCNC.2009.4917944)

Publication date:
2009

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Hu, L., Dittmann, L., & Le Boudec, J.-Y. (2009). Reputation-based content dissemination for user generated wireless podcasting. In *Proceedings, WCNC* (pp. 1-6). IEEE. <https://doi.org/10.1109/WCNC.2009.4917944>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Reputation-based content dissemination for user generated wireless podcasting

Liang Hu Lars Dittmann
Technical University of Denmark
2800 Lyngby, Denmark
{lhua, ladit}@fotonik.dtu.dk

Jean-Yves Le Boudec
Swiss Federal Institute of Technology
1015, Lausanne, Switzerland
jean-yves.leboudec@epfl.ch

Abstract—user-generated podcasting service over human-centric opportunistic network can facilitate user-generated content sharing while humans are on the move beyond the coverage of infrastructure networks. We focus on the aspects of designing efficient forwarding and cache replacement schemes of such service under the constraints of limited capability of handheld device and limited network capacity. In particular, the design of those schemes is challenged by the lack of podcast channel popularity information at each node which is crucial for forwarding and caching decisions. We design a distributed reputation system based on modified Bayesian framework that enable each node estimates the channel popularity in a efficient way. It estimates channel popularity by not only first hand observations but also second hand observations from other nodes. Our simulation result shows reputation system can always well estimate most popular, intermediate and low popular channels, compare to history-based rank scheme which can only well estimate a few most popular channels. Reputation system significantly outperforms history-based rank when the public cache size is small or “a” parameter of Zipf-like distribution is small.

I. INTRODUCTION

In recent years, opportunistic network has become an attractive research area for networking small mobile devices carried by human being, vehicles and animals. Besides unicast routing, dissemination based routing is proposed by PodNet project [1] to provide seamless content distribution beyond infrastructure network. This dissemination based routing particularly support applications in which the set of user interested in receiving a given data is not known in advance. In this paper, we focus on designing reputation-based content forwarding and cache replacement schemes for User Generated Wireless Podcasting (UGWP) service over the system architecture of PodNet. We mainly target at obsolete podcasting service where only the most recent content is of interests and old content is always obsolete by the latest one e.g. short news report distribution or software updates of mobile devices. In UGWP, obtaining popularity information of podcast channels is significant for the content forwarding and cache replacement decisions. Unlike existing Internet-based user generate service such as YouTube [2] where the content popularity information is made centralized, in ad-hoc podcasting, the channel popularity information is fully distributed throughout the network and dynamic due to nodes' mobility. Thus it is much more difficult for each node to obtain

and predict popularity information of global channels. With inaccurate channel popularity information, node may forward the content that future encounter nodes are not interested in. Ultimately, this would lead to low hit ratio of content retrieve, low utilization of both the node contact opportunities and cache storage.

The contributions of this work are two-folds: Firstly, we propose three forwarding and caching replacement schemes and evaluate their performance assuming the ideal knowledge of channel popularity at each node of the network. We aim at studying the optimal forwarding and cache replacement schemes under various scenarios. Secondly, we design a distributed reputation system based on modified Bayesian framework through which each node can efficiently estimate channel popularity. The main idea of our reputation system is as follows: The popularity of channel is represented by the reputation rating. The reputation system consist of three parts: Firstly, the reputation rating of channels at each node is built and updated by the number of requests to each channel from encounter nodes. This is called the first hand information of channel popularity in the sense that it is each node's direct observation. Secondly, reputation rating is also updated by integrating its encounter nodes' direct observations which is called the second hand information of channel popularity. By doing so, node can learn and adjust popularity information of channels from observations made by others even before having to learn by its own experience. Thus, the accurate channel popularity information can propagate much faster throughout the network. Thirdly, to adapt the channel popularity shifts, both the first hand information and the reputation ratings of each channel decays after each contact. The previous observations are gradually forgotten while more weight is put on recently observations.

To the best of our knowledge, our work is the first attempt to employ Bayesian framework based reputation system in the context of information dissemination over opportunistic networks. Previous, the Bayesian framework based reputation system has been employed in coping with misbehaviors in mobile ad hoc networks [3]. The security and cooperation aspects of UGWP are not included in this study. For node cooperation, we assume, to join UGWP service, node is required to contribute a minimum amount of its cache for caching public interested content. Research on opportunistic networks has mainly focused on unicast routing issues so far [4]. Instead, we focus on data dissemination routing to support

applications in which the set of users interested in receiving a given data is not known in advance. There are mainly two classes of data dissemination routing protocols over human-centric opportunistic networks: protocols based on content characteristic (e.g. content popularity, content availability) and protocols based on social characteristics/relations of nodes (e.g. community and centrality of the nodes). The concept of receiver-driven broadcast proposed by Gunnar [5] belongs to class 1 data dissemination protocol. Instead of explicitly pushing public interested content to encounter nodes, each node pulls public interested content from peer node based on own estimated channel popularity and channel solicitation protocols [1]. Yet, the channel popularity is estimated only by node's first hand observations without aging. Along another line, as one example of class 2, [6] propose a socially-aware routing framework for content dissemination in human based opportunistic network. In their work, the focus is to explore the social properties of nodes and identify the best content carrier for the specific content based on the social ties of nodes. Our work focuses on the exploring the popularity of podcast channel, instead of nodes' social ties, thus belongs to class 1 data dissemination schemes. The rest of the paper is organized as follows: Section II describes the concept of modified Bayesian framework based reputation system. Section III describes data structure and protocol specification of reputation system based wireless podcasting. Section IV contains the performance evaluation of forwarding and public cache replacement schemes and Bayesian framework based reputation system. Section V concludes the paper.

II. BAYESIAN FRAMEWORK BASED REPUTATION SYSTEM

A. Standard Bayesian Framework

Node i model the popularity of channel j as an actor in the base system as follows. Node i thinks that there is a parameter θ such that the channel i is interested by any node with probability θ . The outcome is drawn independently from observation to observation (node i thinks there is a different θ for different channel j while different node i may have different believe in different parameter θ). The parameters θ are unknown, and node i model this uncertainty by assuming θ itself is drawn according to a distribution (the "prior") that is updated as new observations become available. We use Beta (A, B) as the prior distribution since it is suitable for Bernoulli distribution and the conjugate is also a Beta distribution. The standard Bayesian procedure is as follows. Initially, the prior is Beta (1,1), the uniform distribution [0,1]; this represents absence of information about which θ will be drawn. Then after ($f+s$) observations during contacts with encounter nodes, say with s times the channel i is requested by encounter nodes while f times it is no requested by encounter nodes. The prior is updated:

$$A := A + s, B := B + f.$$

If θ , the true unknown value is constant, then after a large number m of contacts:

$$A \approx n\theta, B \approx n(1-\theta)$$

and Beta (A, B) becomes close to a Dirac at θ , as expected. We denote $E(\text{Beta}(A, B))$ as the expectation of Beta (A, B). Thus we can estimate θ as follows:

$$\theta \approx E(\text{Beta}(A, B)) = \frac{A}{A+B}$$

B. First hand information by modified Bayesian approach

The first hand information for the popularity of channel j at node i is defined as:

$$F_{i,j} = (A_j^i, B_j^i)$$

This represents the parameters of the Beta distribution assumed by node i in its Bayesian view of the popularity of channel j as an actor in the base system. Initially, it is set to (1, 1). The standard Bayesian method gives the same weight to each observation regardless of its time of occurrence. However, the popularity of a podcast channel may change when nodes move between different communities with different channel popularity distribution. For this reason, we add a reputation fading mechanism to give less weight to the past observations, because the latest observations would be more important for estimating current and future popularity of the channel. Assume node i makes one individual observation of channel j during a contact with encounter node. Let $s=1$ if channel j is requested by the encounter node, and $s=0$ otherwise. The update is as follows:

$$A_j^i := u \bullet A_j^i + s, \quad B_j^i := u \bullet B_j^i + (1-s)$$

The weight u is a discount factor for the past experiences, which serves as the fading mechanism.

C. Reputation Rating and Model Merge

The reputation rating of channel j at node i is defined as:

$$R_{i,j} = (\alpha_j^i, \beta_j^i)$$

Initially, it is set to (1, 1). It is built and updated on two types of events: (1) when first-hand information is updated by own observations; (2) the second hand information from encounter nodes are accepted and copied. There are two variant of using second hand information from encounter nodes: direct observations (first hand information) from encounter nodes and reputation rating from encounter nodes. For event type (1), the update of reputation rating is the same for the first-hand information updating. Let $s \in \{0, 1\}$ is the observations:

$$\alpha_j^i := u \bullet \alpha_j^i + s, \quad \beta_j^i := u \bullet \beta_j^i + (1-s)$$

For the case (2), if we assume passing direct observations, the linear pool model is used to merge own reputation rating with direct observations passed from encounter nodes on the condition if the deviation test is passed. Deviation test is used to protect system against false rating from encounter nodes. The idea behind it is that humans only believe the opinions from others only if, to them, it seems likely i.e. it does not differ too much from their own opinions. Moreover, even if they accepted opinions from others, they only attach less weight to other's opinions than their own opinions. Let

the first hand information of channel j at encounter node x :

$$F_{x,j} = (A_j^x, B_j^x)$$

The deviation test is as follows:

If $|E(\text{Beta}(\alpha_j^i, \beta_j^i)) - E(\text{Beta}(A_j^x, B_j^x))| < \text{THS}$, (THS is a positive constant (deviation threshold)), then the deviation test is passed and we believe the report from node x is trustworthy. Then, α_j^i, β_j^i are updated by first hand observations of node x using the linear pool model merging:

$$\alpha_j^i = \alpha_j^i + w \bullet A_j^x; \beta_j^i = \beta_j^i + w \bullet B_j^x, 0 < w < 1.$$

III. DATA STRUCTURE AND PROTOCOL SPECIFICATION

The cache at each node consists of a private cache (for storing node's private or own interested channels) and a public cache (for storing other nodes' interested channels). Each node maintains a table of channel reputation ratings which is used for content forwarding and public cache replacement decisions. As an example, the reputation rating table of node A is as showed in table 1:

Table 1: Reputation Rating Table

Reputation rating table at node A

Channel Feeds	First Hand Information	Reputation Rating	Latest Entry ID (entry name, time of publish)	Subscribed or Not (S/N)
1	A_1^A, B_1^A	α_1^A, β_1^A	-Weather forecast of Copenhagen -13th July at 10:00 am	S
3	A_3^A, B_3^A	α_3^A, β_3^A	-BBC news -10th July at 10:00 am	S
5	S
7	S
9	S
.....	N
M	A_M^A, B_M^A	α_M^A, β_M^A	N

Channel Feeds: Podcast channel identifications.

A_M^A, B_M^A : the first hand information of channel M at node A.

α_M^A, β_M^A : the reputation rating of channel M.

Latest Entry ID: the latest updates correspond to each podcast channel.

"S": the channel is subscribed by node A.

"N": the channel is not subscribed.

In brief, the protocol specification of reputation system based podcasting is as follows:

1. Idle node periodically broadcast association requests to its neighbors. If it discovers several neighboring nodes, it randomly selects one node to associate and establish pair-wise connection.
2. Node updates its reputation ratings of all channels by merging the second hand information from peer if the deviation test is passed.
3. Node firstly pulls content of private interested channels.
4. Node updates both first hand information and reputation rating of channels by peer's requests of privately interested channels.
5. Node pull content of public interested channels based on estimated channel popularities and popularity-based forwarding and public cache replacement schemes. Various forwarding and public cache replacement schemes are described below.

Public interested content forwarding scheme:

Most (M): Based on node's own channel popularity estimation, node firstly forward the content of the most popular channel from its peer node if there is new updates, then the second most popular one, the third most popular one

and so on, until the association of two nodes breaks either when they move apart from each other or the content download process finishes. The aim of forwarding most popular channels first is to maximize the probability that future encounters would be interested in requesting it.

Probabilistic (P): node decides to forward a public interest channel with a probability proportional to its popularity (by the node's local estimation). This scheme gives most network capacity to most popularity channels while still gives certain network capacity to intermediate and low popular ones.

Uniform (U): A node decides which channels to forward content with equal probability. Thus the network capacity is evenly given to all the channels. Thus, node does not need to estimate the popularity information of channels for forwarding decisions.

Public cache replacement scheme:

When the public cache of a node is full and there are new public interests channels at peer node, one has to decide whether to replace public interests channel already in the public cache with new public ones from peer. If it decides so, it also needs to decide which public interests channels to replace.

Most (M): Only if the channel from peer is at least more popular than the least popular public interests channel in the public cache, the node can forward this new channel. If so, the least popular channel in public cache will be replaced by this new public interests channel from peer. The channel popularity is based on the node local popularity estimation.

Probabilistic (P): When public cache is full, whenever there are updates of any public channels, node deleted the content of the channel in the public cache with a probability which is proportional to its popularity (based on node local rating table).

Uniform (U): When public cache is full, whenever there are new updates of public channels, node deleted the content of the channels in the public cache with equal probability. Nodes do not need to have the channel popularity information.

IV. PERFORMANCE EVALUATION

In this section, we firstly compare various forwarding and cache replacement schemes under the ideal knowledge of channel popularity information at each node. Then we evaluate the performance of reputation systems by comparing it with a history-based rank popularity learning scheme [1].

A. Simulation Settings

The performance evaluation is done with our own simulator which is based on a simple communication model: two nodes can communicate with a nominal bit-rate if their geometric distance is smaller than a threshold value (that models the radio range of mobile device). The simulation model does not incorporate link layer issue such as collision or interference, since we simulate a sparsely connected network where the collisions or interference among different associations are very rare. For the simulation, we further assume that the setup time for nodes' pair-wise associations is 12 second which includes neighbor discovery time and node synchronization time [7].

We assume a scenario where human beings carry Wi-Fi enabled mobile portable device. For that purpose, we set nodes move with a constant moving speed 1m/s (average human walking speed) and constant pause time 1s according to Random Way Point (RWP) mobility model. The radio range of each device is 38 meters (in door wireless range of 802.11b) and the nominal rate of the radio device is 4.5 Mbits/s. We further assume in total 100 nodes uniformly distributed in a square with diameter (1500 m, 1500 m). Nodes only associated pair-wise, even if more than two are within reach of one another. The reason is that the contact duration may be short and it is better to get high throughput by only sharing the transmission capacity between two parties than to get high connectivity. Each node can publish one channel to other nodes of the network, but it is not mandatory. For simplicity, we also assume each node generate new contents from its channel periodically in identical time interval e.g. every 300 second. Besides publishing content, each node is interested in two channels published from other nodes. The global popularity distribution of podcast channels follows Zipf-like distribution. We assume the lower the channel index, the higher the popularity, i.e. channel 0 is the most popular one, channel 1 is the second most popular and so on. Thus, the popularity of channel i is given as follows:

$$P_i \sim \frac{1}{(i+1)^a}, i = 0, 1, 2, \dots, 99$$

Each node has 2G bytes cache which consists of public cache and private cache. Each data chunk is 2M byte, thus downloading one chunk takes 4s with pair-wise association and 802.11b MAC. One chunk is assumed to be complete and atomic unit and thus has no relation to other chunks. For example, it could be 10 minutes audio of BBC news as a part of the whole 60 minutes BBC news program. The semantic of podcasting service is assumed to be obsolete, where only the most recent chunk of each channel is kept in the cache. For a given channel, once new chunk of that channel is received, the old chunk would be immediately deleted. However, each node can optionally keep its own interested chunks in private cache. The total simulated time is 12 hours. The simulation parameters of reputation system are THS=0.4, $u=0.99$, $w=0.2$.

B. Performance Metrics

To quantify the user satisfaction of user generated podcasting, the recall and delay are employed as the performance metrics of reputation system. Recall is defined as the fraction of node's own interested chunks that are successfully received. It is borrowed from the area of Information Retrieve (IR). Delay is defined as the latency between the time when chunk is published and the time when it is received. We believe, for obsolete pocast service, both recall and delay are important for the end user satisfaction. Recall of node i by time t is defined as:

$$R^i(t) = \frac{X_R^i(t)}{X_P^i(t)}, i = 0, 1, 2, \dots, N-1$$

N : the total number of nodes; i : the node ID. $X_R^i(t)$: the total number of private interested chunks that have been received

by node i by time t . $X_P^i(t)$: total number of private interested chunks that have been published from all node i 's interested channels by time t . Average recall is defined as the average recall over the total number of nodes N . In this work, we are only interested in the average recall at the end of the simulation. Delay is defined as $\Delta t = T_{publish} - T_{receive} \cdot T_{publish}$ is the chunk publish time while $T_{receive}$ is the time when it is received. M is defined as the total number of chunks received by all nodes at the end of simulation. The average delay is defined as:

$$\frac{\sum_i \Delta T_i}{M}, i = 1, 2, 3, \dots, M$$

C. Simulation Results

1. Comparison of forwarding and cache replacement schemes under the ideal knowledge of channel popularity

We assume all nodes have prior knowledge of the global channels popularity information and their interested channels out of all the channels. We compare the performance of three combinations of public content forwarding and public cache replacement schemes as defined in table 3.

Table 3: Simulation Parameters

Zipf-like distribution	publish interval	Public cache	Number of channels
a=1.0	300 s	30 chunks	10, 20, 50, 100

As showed in fig 1 and 2, When the number of channel is small (e.g. 10, 20), all the schemes achieve identical performance of both average recall and average delay. As the number of channels increases, MM and PP performs much better than UU. In particular, when the number of channel is 100, MM can outperform UU almost 100% of average recall and 600 second of average delay. The reason is as follows: for a given channel popularity distribution and fixed number of nodes, when the number of channels is small, all channels are very popular among the nodes. It does not matter how network capacity and public cache capacity is allocated to different channels (according to one specific forwarding and cache replacement scheme). Forwarding and caching any channel would bring a high hit rate from the future encounter nodes. Thus, MM, UU, and PP perform similar in this case. However, as number of channel increases, the number of unpopular channels increases. In this case, the allocation of network capacity and public cache capacity do matters. With UU scheme, too much network and public cache capacity would be wasted for forwarding and caching unpopular channels which are rarely requested; In contrast, popular channels being highly requested cannot get sufficient network resources. MM can more efficient utilize network resources than UU by allocating most network resources to popular channels which are highly requested and least capacity to unpopular channels which are rarely requested. Thus, MM and PP significantly outperforms UU when the number of channel is large.

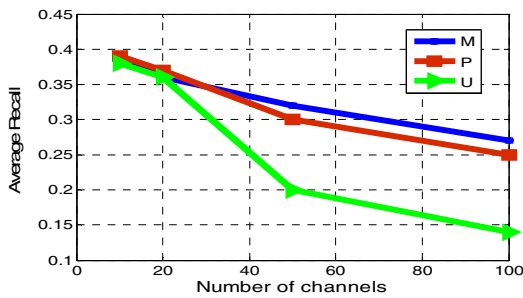


Figure 1: Average recall under various numbers of channels

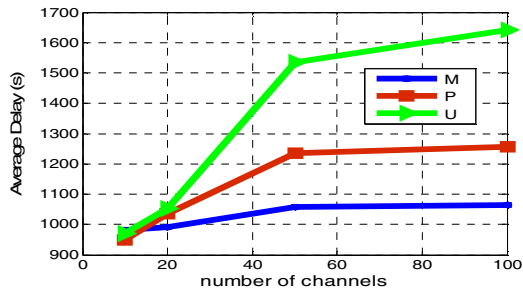


Figure 2: Average delay under various numbers of channels

2. Performance evaluation of modified Bayesian framework based Reputation system

In realistic case, however channel popularity information is not ideally known to each node. In this section, assuming Most-Most scheme is employed, we evaluate the performance of reputation system by comparing it with history-based rank scheme [1]. With history-based rank estimate, channel popularity is estimated only by node's direct observation represented by number of requests per channel from encounter nodes. Typically, node keeps track of the channels that were requested by past encounter nodes and maintains a history-based ranking. Only the requests for the channels of encounter nodes' own interests are counted. We firstly compare the channel popularity evolution over time at node 5 (node ID) for the two channel popularity estimation methods. The channel popularity is represented by the number of requests from encounter nodes and by reputation ratings respectively. Secondly, we compare network performance of the two methods, under the impact of various public cache sizes and "a" parameter of Zipf-like distribution. The simulation parameters are in the table 4:

Table 4: Simulation Parameters

Zipf-like distribution	Number of channel	Public cache size	Publish interval
1.5	100	30 chunks	300 s

Fig 3 shows the performance of history-based rank scheme in channel popularity estimation at node 5. Subsets of the channels' popularity information are showed, in particular channel 0, 1, 3, 12, 20, and 50. The vertical axis is the number of requests per channel from node 5's encounter nodes while the horizontal axis is time (unit is two minutes). We observe that the high popular channels (e.g. channel 0, 1, and 3) can be accurately estimated from the start to the end of the simulation. However, the intermediate and low popular channels (e.g. channel 12, 20, and 50) are not well accurate

until a long simulated time has past. There are no observations of popularity information of that channel for a very long simulated time. Take channel 12 for example: only after 264 minutes, node 5 starts to get the popularity information of channel 12. The reason is that, only by node 5's direct observation, it takes a very long time to collect the popularity information of intermediate and low popular channels since there are no requests of those channels at node 5 for a long simulated time. In other words, due to the lack of the direct observations before time 264 minutes, node 5 would consider channel 12, 20, 50 and 80 as the same popular channels. This can negatively influence the forwarding and cache management decision.

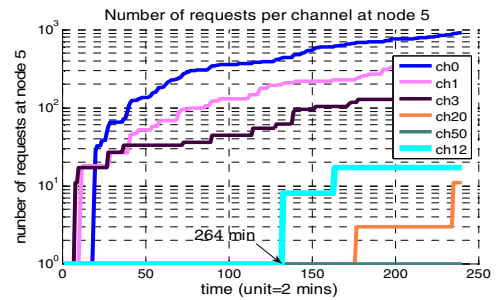


Figure 3: History-based rank: number of requests per channel

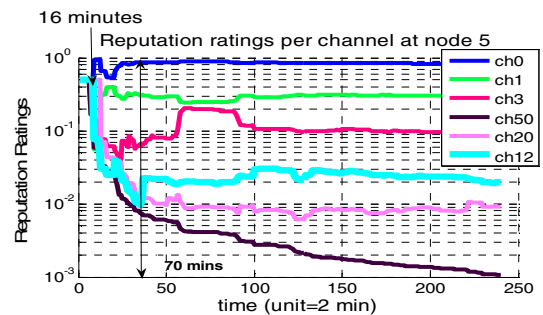


Figure 4: Reputation system: reputation ratings evolution

Figure 4 shows reputation system can accurately estimate the popularity of both high popular channels and low popular ones already from the start of the simulation. Though the reputation ratings slightly fluctuate in the initial phase of simulation, they get stable very fast. Even if there are not enough direct observations for estimating low popular channels, node can still make use of second hand information from encounter nodes to have a more accurate and faster estimation than history-based rank method.

Next we compare the performance of reputation system with history-based rank under the impact of public cache size and "a" parameter of Zipf-like distribution. We also use the MM scheme under ideal knowledge of channel popularity as the optimal performance baseline. In terms of average recall, reputation system always performs better than history-based rank scheme under various public cache sizes, as showed in figure 5. Especially when the public cache size is small, reputation system can overwhelmingly outperforms history-based rank. In this case, reputation system can outperform

100% over history-based rank when the public cache is 5 chunks. As the public cache decreases, the performance of history-based rank drops dramatically. The reason of this trend is that history-based rank performs worse as public cache size decreases. Smaller public cache size indicates fewer chunks are likely to be requested per time unit by the encounter nodes. A smaller number of chunks requested by encounter nodes would result in smaller amount of first hand information per time unit, which ultimately brings lower performance of history-based rank.

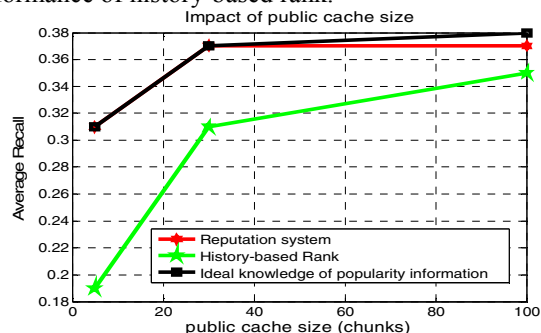


Figure 5: Average recall under various public cache sizes

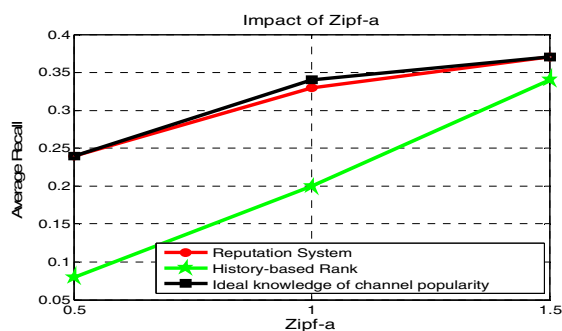


Figure 6: Average recall under various "a" parameters

Fig 6 shows, under the impact of Zipf-a, the performance of history-based rank scheme drops more dramatically than reputation systems in terms of average recall. It indicates that history-based method performs worse as the "a" parameter of Zipf-like distribution becomes smaller. The analysis is as follows: for a given Zipf-like distribution, accurate estimations of most popular and intermediate popular channels are both important for the network performance, while low popular ones are not as important as they are rarely requested. History-based rank can only estimate a few most popular channels. When "a" parameter is large e.g. 1.5, there are only most popular channels and low popular ones, with only few intermediate popular ones. History-based method's performance can approach two variant of reputation systems by well estimating most popular channels. When "a" parameter decreases from 1.5 to 0.5, the number of intermediate popular channels increases while the number of most popular ones decreases. In this case, the performance of history-based rank becomes worse since more intermediate popular channels cannot be accurately estimated due to the lack of popularity information by direct observations. More intermediate channels get as few forwarding opportunities as

low popular channels do, since they are estimated to be equally popular. On the other hand, the performance of reputation system is less sensitive to the "a" parameters, with only small performance decrease when "a" parameter becomes small. By taking account both direct observations and second hand observation, it can always well estimate both most popular channels and intermediate popular ones for any "a" parameters.

V. CONCLUSION

We aim at designing a reputation-based user-generated wireless podcasting service over human-centric opportunistic network. Firstly, we propose various forwarding and public cache replacement schemes under the ideal knowledge of channel popularity at each node. Simulation results shows that when the number of channel is large, MM schemes performs best while UU performs worst in both average recall and average delay; On the other hand, when the number of channel is small, the difference of various schemes is minor. Secondly, we propose a modified Bayesian framework based reputation system for estimate the channel popularity. By both first hand observations and sharing second hand observations with other nodes, node can obtain the channel popularity information much faster and more accurate. Simulation results shows reputation system can always well estimate most popular, intermediate and low popular channels, compare to history-based rank which can only well estimate a few most popular channels. Reputation system can significantly outperforms history-based rank when the public cache size is very limited (e.g. 5 chunks) or "a" parameter of Zipf-like distribution is small (e.g. between 0.5 and 1).

REFERENCES

- [1] Vincent Lenders, Martin May, and Gunnar Karlsson. Wireless Ad Hoc Podcasting. In Proceedings of IEEE SECON, San Diego, CA, June 2007.
- [2] <http://www.youtube.com/>.
- [3] Sonja Buchegger, Jean-Yves Le Boudec. Self-Policing Mobile Ad-hoc Networks by Reputation Systems. Communications Magazine, IEEE Volume 43, Issue 7, July 2005.
- [4] Luciana Pelusi Andrea Passarella Marco Conti. Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. Communications Magazine, IEEE. Publication Date: November 2006 Volume: 44, Issue: 11
- [5] Gunnar Karlsson, Vincent Lenders, Martin May. Delay-Tolerant Broadcasting. IEEE Transactions on Broadcasting, vol.53, NO.1, March 2007.
- [6] Costa, P. Mascolo, C. Musolesi, M. Picco, G.P. Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. Selected Areas in Communications, IEEE Journal on Publication Date: June 2008 Volume: 26, Issue 5.
- [7] Anna-Kaisa Pietilainen, Christophe Diot. Experimenting with real-life opportunistic communications using windows mobile devices. International Conference On Emerging Networking Experiments And Technologies Proceedings of the 2007 ACM CoNEXT conference.